Vision-based Space Autonomous Rendezvous : A Case Study

Antoine Petit, Eric Marchand, Keyvan Kanani

***Abstract*— For a space rendezvous mission, autonomy imposes stringent performance requirements regarding navigation. For the final phase, a vision-based navigation can be a solution. A 3D model-based tracking algorithm has been studied and tested on a mock-up of a telecommunication satellite, using a 6-DOF robotic arm, with satisfactory results, in terms of precision of the pose estimation and computational costs. Quantitative tests in open loop have been carried out to show the robustness of the algorithm to relative inter frame motions chaser/target, orientation variations and illumination conditions. The tracking algorithm has also been successfully implemented in a closed loop chain for visual servoing.**

I. INTRODUCTION

A. Context : an autonomous space rendezvous mission

A space rendezvous consists in the approach of a chaser spacecraft towards a target spacecraft, from detection of the target (if necessary) until docking on the target [15], [10]. It can be achieved manually or autonomously. A high level of autonomy can be preferred for several reasons (ground control cannot be used because of large communication delays, safety reasons...). This maneuver has various applications, from space station supplying, spacecraft refueling, grasping or repairing. The first rendezvous was done manually in 1966, between Gemini 8, commanded by Neil Armstrong, and Agena target vehicle. More recent examples include the European Space Agency (ESA) Automated Transfer Vehicle (ATV) which autonomously docks onto the International Space Station (ISS), and projects such as the ESA Geostationary Servicing Vehicle (GSV) or the RObotic GEostationary orbit Restorer (ROGER), both tasked to capture, inspect, assist or re-orbit satellites in trouble. The following main phases make up a rendezvous mission (Fig. 1):

1) the launch of the chaser (several days), which reaches the target orbital plan (A, B), in absolute navigation, and detection of the target.
2) the intermediate phase (a few hours until a day), in which the chaser approaches the target orbit, in absolute navigation. It consists in orbital transfers such as Hohmann transfers (B to E).
3) the terminal phase (a few orbit periods) consists in getting closer to the target down to a secure distance, in relative navigation, with successive impulsive orbital maneuvers. Depending on the mission (ATV, Soyouz/Progress, DART), these maneuvers are handled differently [15].
4) a forced translation is performed during a few minutes to dock onto the target.

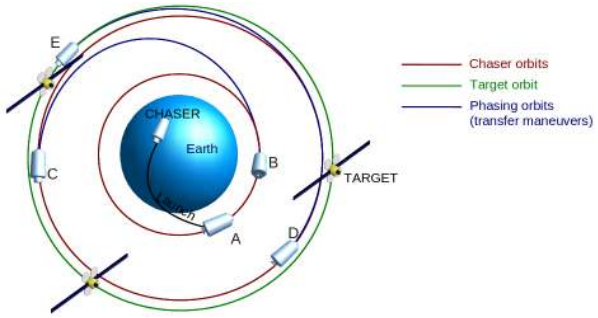


Fig. 1: Phases of a typical rendezvous

B. Scope of the study

In this work, which extends a former study performed by Astrium known as High-Integrity, Autonomous, Multi-Range Rendezvous and Docking (HARVD) [14], the target is a telecommunication satellite (Amazonas-2). Since such a satellite can be tracked with good accuracy from the ground, the chaser can be driven on the target orbit and quite close to this target (distance $< 50km$). For the autonomous terminal phase, the secure distance is set to $1000m$ [14]. Along with the forced translation, an alignment phase, or *fly-around* phase (Fig. 2), with the target docking port is introduced. We only focus on the final approach phase, the most critical one, with precise requirements regarding lateral, longitudinal and angular alignment and computational costs, and whose navigation is proposed to be achieved by computer vision (along with inertial sensors), using a monocular camera mounted on the chaser, for a distance ranging from $1000m$ to contact.

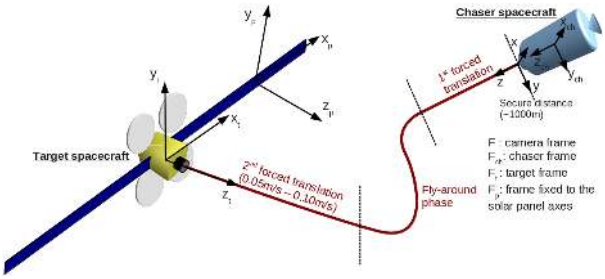


Fig. 2: Final approach of a rendezvous mission

A. Petit is with INRIA Rennes - Bretagne Atlantique, Campus de Beaulieu, 35042 Rennes, France, Antoine.Guillaume.Petit@inria.fr

E. Marchand is with Université de Rennes 1, IRISA, Campus de Beaulieu, 35042 Rennes, France, Eric.Marchand@irisa.fr

Keyvan Kanani is with Astrium Satellites, 31 Avenue Des Cosmonautes, 31402 Toulouse, France

C. State of the art

Navigation of the chaser consists in continuously estimating its relative pose (position, orientation) with the target. To address this task, some computer vision approaches have already been implemented for proximity operations in a rendezvous mission. On the ATV, a videometer processes a laser beam reflected by retroreflectors installed on the ISS, which is the first operational vision system for spacecraft navigation [13]. Using cameras, [1], [2], [15] propose to rely on easy to detect and track markers installed on the target. Other techniques under study in a space context deal with natural features with stereo vision methods to recover the pose, given a 3D model of the object [8], [11]. With a monocular camera, a feature-matching computer vision approach had been selected in [14]. It consists in the extraction of invariant features in the image that are matched to a database built from preliminary learning sessions. It is however too computationally intensive to be used during the whole mission at the required frequency (a few Hz). A critical sensitivity to distance, illumination, relative orientation, and occlusions of the target had been observed [14]. In this paper we apply a markerless 3D model-based tracking algorithm to the problem of satellite tracking using a monocular camera [6]. It has already been implemented in an ESA aerospace applications [8] and in an aeronautics context [7].

The tracking algorithm is recalled in Section II. In Section III, we present a visual servoing control loop to automatically perform a simulated rendezvous between a robot and a mock-up of Amazonas-2. Results regarding tracking performances and robustness in open loop and regarding visual servoing are exposed in Section IV.

II. 3D MODEL-BASED TRACKING

Our problem is restricted to model-based tracking, using a 3D model of the target. This model is made up of lines, which provides a good invariance to pose and illumination changes, and robustness to some image noise or blur. The purpose is to compute the pose (camera, and thus chaser, position and orientation with respect to the target) which provides the best alignment between edges of the projected model and edges extracted in the image [12], [9], [6]. The approach considered in this paper is described in [6]. Given a new image, the 3D model of the scene or the target is projected in the image according to the estimated previous camera pose $\mathbf{r}$ (see Fig.3). Each projected line $L_i(\mathbf{r})$ of the model is then sampled along its orientation in the image with a sample step S. Then from each sample point $p_{i,j}$ a 1D search along the normal of the projected edge is performed, for a range R, to find a corresponding point $p'_{i,j}$ in the image. As in [6], $p'_{i,j}$ is of maximum likelihood with regard to $p_{i,j}$.

In order to compute the new pose, the distances between points $p'_{i,j}$ and the projected lines L_i are minimized with respect to the following criteria [6] :

$$\Delta = \sum_i \sum_j \rho(d_\perp(L_i(\mathbf{r}), p'_{i,j})) \quad (1)$$

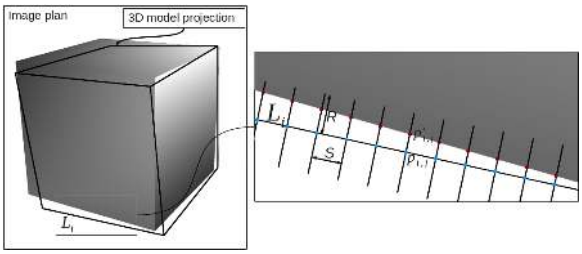


Fig. 3: Moving Edge principle

where $d_\perp(L_i(\mathbf{r}), p'_{i,j})$ is the distance between a point $p'_{i,j}$ and the corresponding line $L_i(\mathbf{r})$ projected in the image from a pose $\mathbf{r}$. Here, ρ is a robust estimator, which reduces the sensitivity to outliers. This is a non-linear minimization process with respect to the pose parameters $\mathbf{r}$. The minimization process follows the Virtual Visual Servoing framework [6] similar to a Gauss-Newton approach. The tuning of the range R, sample step S, and the maximal number of iterations K in minimization process are discussed in Section IV, regarding estimation precision and computational costs.

III. TRACKING FOR VISUAL SERVOING

In order to servo the robot, we propose to use the 3D-model based tracking algorithm within a 2 1/2 D visual servoing control loop.

Visual servoing consists in using data provided by a vision sensor for controlling the motions of a dynamic system [3]. Classically, to achieve a visual servoing task, a set of visual features $\mathbf{s}$ has to be selected from the image enabling the control of the desired degrees of freedom. The goal is to minimize the error between the current values of visual features $\mathbf{s}$ extracted from the current image and their desired values $\mathbf{s}^*$. For this purpose, techniques [3] depend on the features $\mathbf{s}$ used : they can be 2D points directly extracted from the image, for Image-based Visual Servoing (IBVS) or 3D parameters recovered thanks to image measurements like pose computation for Position-based Visual Servoing (PBVS). Here we apply a hybrid solution, 2 1/2 D visual servoing approach [4], [5], which avoids the shortcomings of the two basic approaches, by combining features in 2D and 3D, in order to decouple position and rotational movements, with a simpler interaction matrix, and with a better stability than IBVS or PBVS :

$$\mathbf{s} = \begin{bmatrix} x & y & \theta u_z & \mathbf{t} \end{bmatrix}^T \quad (2)$$

where x and y are the metric coordinates in the image of a point of the object, here the center of the mock-up, θu_z is the third coordinate of the θu vector, which represents the rotation the camera has to perform to reach the desired pose, and $\mathbf{t}$ is the translation vector the camera has to perform to reach the desired pose, expressed in the desired camera frame. θu_z and $\mathbf{t}$ have thus to be regulated to 0. We need to minimize the error $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$ where features $\mathbf{s}$ are recovered thanks to model-based tracking. A kinematic controller, which is convenient for most of systems, is then designed to servo the camera. A proportional control scheme

is defined, to make the error exponentially decreases, leading to the following control law:

$$\mathbf{v_c} = -\lambda \widehat{\mathbf{L_s}}^{+}(\mathbf{s} - \mathbf{s}^*) \quad (3)$$

with $\widehat{\mathbf{L_s}}^{+}$ the estimate of the pseudo-inverse of $\mathbf{L_s}$, the interaction matrix associated to the visual features. The paper described in [3] details how this matrix can be computed. It can here be estimated thanks to the parameters of the pose computed by the model-based tracking algorithm.

IV. RESULTS

A. Experimental conditions

To implement the tracking algorithm on a vision based rendezvous context, Astrium provided a complete 3D-model and a real reduced (1/50) mock-up of Amazonas-2, a telecom satellite built from the Eurostar-3000 platform, and similar to the one used for HARVD experiments [14]. Amazonas-2 was launched in 2009 for Spanish company Hipsasat to cover the American (especially South America) position. It is located on a Geostationary Orbit.

The provided 3D model is too complex to deal with real-time applications. Thus we have considerably simplified the model, keeping the most significant geometrical features. Besides, depending on the relative size of the mock-up in the image, the relevant information to be considered varies. Indeed, regarding the central module of the satellite, the contours are not very precise and regular, due to the insulating film on most of the surface of this module and on the four circular antennas. Since the film reflects light, the irregularities are enhanced by the sun luminosity and make it hard for the algorithm to identify edges. For long distances (Fig.6(a)), the information provided by the central module is not useful, nor significant, compared to the information given by the solar panels, with sharper edges. However, for shorter distances (Fig.6(d)), details of the central module have to be included and the model has to be refined, to perform the approach properly. The solution implemented here is to switch between 3D models in order to use the model with the most relevant information with regard to the distance camera/mock-up.

Using a robot to simulate the rendezvous, with a camera mounted on its end-effector, enables to have regular and quite realistic movements (let us however note that the specific dynamic of spacecraft is not considered in this paper). Besides, the position of the robot is known very precisely (10^{-4}m), providing us with ground truth. It enables to exactly replicate test procedures and so to carry out systematic tests to quantify the tracking performances in open loop. The robot we have used (Fig. 4) has 6 degrees of freedom and 640×480 images are processed. Sun illumination can also be simulated by spot lights located around the scene. As the mock-up size is 1/50 of the satellite and as we use for tests a different camera from the one that would be on-board (in this case, a camera with a 5 deg. FoV, along $\mathbf{x}$ and $\mathbf{y}$, is supposed to be mounted on the chaser spacecraft), scaling factors need to be derived. Fig.5 shows the equivalence between the real and experimental contexts with regard to the $\mathbf{z}$-coordinate of the target center in the camera frame, so that the target has the same relative size in the image in both cases. The dynamics along $\mathbf{x}$ and $\mathbf{y}$ are also affected as relative motions chaser/target in the real context scaled by a factor close to 1/50 induce in the experimental context the same inter frame motions of the target in the image. Typical forced translations for rendezvous scenarios specified in [14] are performed with velocities ranging from $0.05m/s$ to $0.10m/s$, and with a camera frame rate set to 1Hz in these cases.

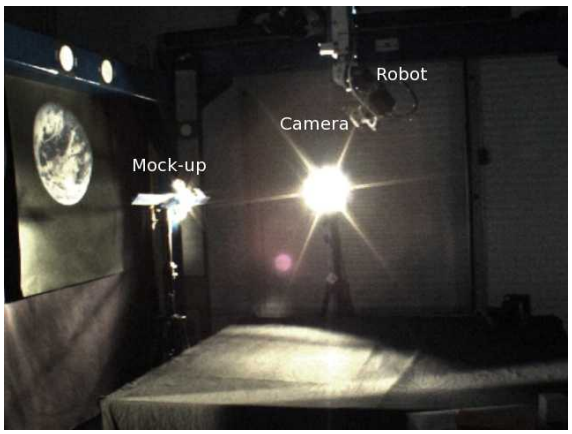


Fig. 4: The experimental setup. It includes Amazonas 2 mock-up, camera mounted on the robot end-effector and a light.

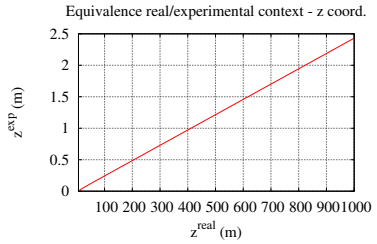


Fig. 5: Equivalence experimental/real contexts. The scaling factor $\frac{dz^{exp}}{dz^{real}}$ can be approximated to 0.0024 as we restrict to distances ranging from $56m$ to $800m$, due to the robot joint limits.

B. Results on a nominal scenario

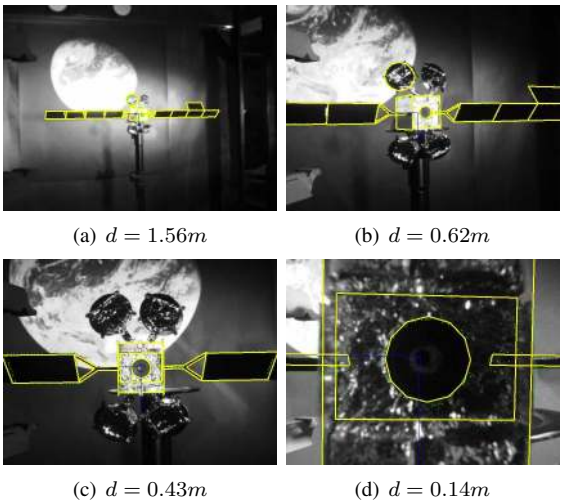
In this scenario, for the first translation, the camera is moving with a constant velocity ($0.015m/s$) along its optical axis, which targets the center of the mock-up, with a 15Hz frame rate (the equivalent velocity would be $0.40m/s$ in a space context, so larger than usual translation velocities), and with a distance chaser/target ranging from $1.56m$ to $0.62m$, what corresponds to a $642m$-$255m$ range in the space context. Then the robot performs a fly-around phase to realign to the docking port of the mock-up and performs its final translation with a constant velocity (still $0.015m/s$), until a distance chaser/target of $0.135m$, which is the shortest achievable distance because of joint limits between the robot and the mock-up, and which is equivalent to $56m$ in the space context. Here, illumination conditions are favorable, leading to distinctive edges on the mock-up. Two tracking configurations, C1 and C2, are implemented, and are specified by

TABLE I: Tracking configurations

	C1	C2
Model	Single model	2 models
d_s (m)	No	0.43
R (pixels)	6	6 ($d > d_s$), 11 ($d < d_s$)
S (pixels)	5	10 ($d > d_s$), 8 ($d < d_s$)
K	30	5

the type of 3D model(s) used, their switching distance d_s if it is the case, and the tracking key parameters defined in Section II (range R, step S, and the maximum number K of iterations in the control law), (see Tab. I). These values have been empirically determined.

Performances are satisfactory for both cases (see Fig. 6 and Fig. 10), with low positioning errors along $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$. In the equivalent space context, these estimation errors along $\mathbf{x}$ and $\mathbf{y}$ would be less than 0.08% of the distance chaser/target, except for C2 from $82m$ ($90s$ on Fig.10) to the center of the target, with around 0.3% of the relative distance, along $\mathbf{x}$. They would always be less than 0.5% along $\mathbf{z}$. Low angular misalignment is also observed (less than 5 deg., except around $\mathbf{y}$ for C2 at the far end of the estimation). In this scenario, solutions C1 and C2 appear to have quite similar performances, except that C1 is more stable than C2, due to its higher level of detail and its smaller sample step S.



(a) $d = 1.56m$ (b) $d = 0.62m$

(c) $d = 0.43m$ (d) $d = 0.14m$

Fig. 6: Tracking for a nominal scenario, under good illumination, with C2. On (a), the chaser starts its first translation, then its fly-around phase on (b). During the second translation, the 3D model and tracking parameters change (c), until $d = 0.14m$ (d).

C. Robustness tests

1) Robustness to translation motions: As the target may not be initially centered in the image or because of uncertainties in the chaser and target motions, it has been relevant to test the sensitivity of the algorithm to translation motions of the robot along $\mathbf{x}$ and $\mathbf{y}$ in the camera frame, with relative orientation alignment. The purpose is to determine

TABLE II: Robustness to motions

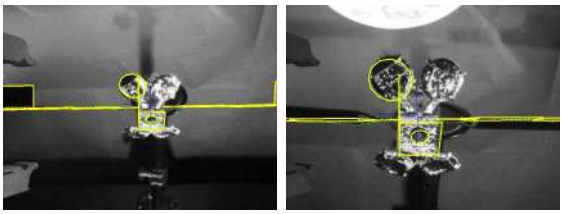
	C1				C2			
Distance (exp) (m)	0.3	0.5	0.9	1.3	0.3	0.5	0.9	1.3
Distance (real) (m)	124	206	370	535	124	206	370	535
Target motion lim. /x (pix/frame)	5.5	6.0	6.6	6.8	8.1	6.0	5.8	5.4
Target motion lim. /y (pix/frame)	3.8	5.4	5.8	6.0	7.9	5.9	5.8	6.0
Rel. motion lim. /x (real) (cm/frame)	9.4	17.0	33.2	50.1	13.8	16.9	29.6	39.8
Rel. motion lim. /y (real) (cm/frame)	8.7	20.4	39.5	59.0	18.0	22.3	39.5	59.0

if at different distances chaser/target the tracking is properly performed or not, given the inter frame motion constraints. Tab. II gathers the results, with limits in terms of the motion of the target in the image and of the relative motion chaser/target in the real space context.

The tracking is less robust to inter frame translation motions of the target in the image at short distances and for motions along $\mathbf{y}$. C2 appears to be more robust than C1 for distances inferior to d_s as its range R is larger. Nevertheless, limits in terms of inter frame relative motions in the space context show that the tracking algorithm is more than suitable for a rendezvous approach with slow and smooth motions, as defined in [14].

2) Robustness to orientation variations: Tests have shown that with C1 or C2 configuration, tracking can be performed, whatever the initial relative orientation. Some problems may occur when the solar panels tend to appear as single edges in the image (see Fig. 7(a)). This singularity leads to some ambiguities on the contours of these panels when the chaser performs its first translation and fly-around phases (Fig. 7(b), Fig. 7(c)). Fig. 11 shows performances of this approach. We observe greater but still acceptable position errors. In the equivalent space context, the error along $\mathbf{z}$ would be below 0.7% of the relative distance for C1, 1% for C2 during the first translation ($0s - 7s$) and fly-around phases ($7s - 20s$) (Fig. 11), then decreasing to remain under 0.36% until the end (at $56m$ from the target in a real space context), in both configurations. Along $\mathbf{x}$ and $\mathbf{y}$, errors would always be below 0.25% of the relative distance for C1 and C2. The orientation error around $\mathbf{z}$ reaches 13 deg. for C1 during the fly-around phase, but then gets much lower. More generally, tracking is limited to rotational motions of the chaser in the camera frame implying inter frame target motions in the image similar to the limits defined in Tab. II.

3) Robustness to illumination changes: A quite extreme scenario has been experimented, with brutal changes (Fig. 8). Fig. 12 shows performances of two solutions, different from C1 and C2. Low luminosity triggers the disappearance of the panels. With the lack of this key geometrical information, the four circular antennas become necessary for the tracking to be performed. The information being concentrated in the middle of the image, we can observe some errors for rotations around $\mathbf{x}$ and $\mathbf{y}$, and of position for $\mathbf{z}$ at the be-

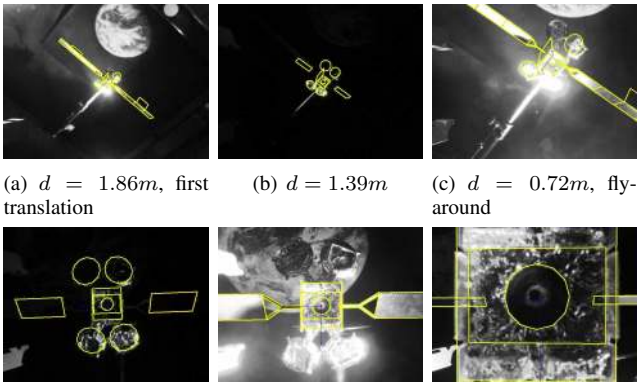


(a) $d = 0.70m$, first translation (b) $d = 0.45m$, fly-around

(c) $d = 0.45m$, fly-around (d) $d = 0.45m$, end of fly-around

Fig. 7: Potential problem when tracking with singular orientation, when the solar panels of the target appear as single edges (a).

ginning. Performances are better for a multi-model solution (two models, switching six times, at illumination changes), especially on the lateral position alignment. The different illumination changes trigger jumps in the different measured errors, see Fig. 12. The tracking is particularly robust to light reflections on the insulating film.



(a) $d = 1.86m$, first translation (b) $d = 1.39m$ (c) $d = 0.72m$, fly-around

(d) $d = 0.48m$, second translation (e) $d = 0.38m$ (f) $d = 0.14m$

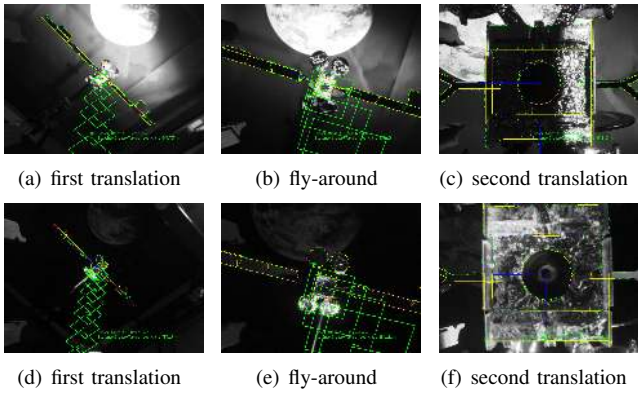
Fig. 8: Tracking with 3D models switching at illumination changes.

D. Computational costs

In our application, the ability to switch between different models and so to adapt them to the conditions (illumination, distance) is a relevant issue to reduce computational costs. The simple 3D models used in C2 have thus been designed in this sense, what has lead to lower computational costs than for C1 and its single detailed model, with more lines. For parameters K, R and S defined in Section II and which tune the estimation convergence rate, C2 is a trade-off to reduce costs while preserving performances as compared to C1. With an Intel Core 2 Duo processor, the mean execution time per frame for a typical approach is $45ms$ for C1, $21ms$ for C2, with a $18ms$ standard deviation for C1, $4ms$ for C2.

E. Visual servoing results

We have performed an autonomous rendezvous in a closed loop chain by servoing the robot close to the mock-up, with angular alignment, using the 2D 1/2 visual servoing technique presented in Section III. As exposed in Section I-B, the maneuver has been divided into three phases: a first translation is achieved to drive the target into the center of the image, a fly-around phase to align to the docking port axis of the target, and a final translation until almost docking ($0.13m$ or $55m$). The servoing performed on the mock-up has successfully achieved the intended goal, for two illumination conditions (Fig. 9(a)-(c) and (d)-(f) and provided video).



(a) first translation (b) fly-around (c) second translation

(d) first translation (e) fly-around (f) second translation

Fig. 9: Complete approach within a visual servoing experiment under strong and low luminosity.

V. CONCLUSIONS AND FUTURE WORKS

A real-time model-based based approach has been applied on the final approach of an autonomous rendezvous mission, in order to track and automatically approach the target spacecraft. The tests performed in open loop on a mock-up have shown promising performances in terms of pose computation, regarding motions, distance, orientation or illumination constraints. The estimation precision and the robustness of the algorithm would make it suitable for equivalent realistic rendezvous final approaches. This method is also adapted to perform an autonomous rendezvous only based on visual servoing, with successful results.

As the tracking method only deals with simple line models, with a simple hidden-face algorithm, future works would aim at processing a complete, complex surface 3D model, thanks to hardware graphics. Evaluating and propagating tracking uncertainty using statistical methods and improving initialization techniques would also be key issues.

REFERENCES

[1] F. Blais, J.A. Beraldin, L. Cournoyer, I. Christie, R. Serafini, K. Mason, S. McCarthy, and C. Goodall. Integration of a tracking laser range camera with the photogrammetry based space vision system. In *SPIE Aerosense*, volume 4025, page 219, April 2010.

[2] Marchall Space Flight Center. Dart demonstrator to test future autonomous rendezvous technologies in orbit. FS-2004-08-113-MSFC, September 2004.

[3] F. Chaumette and S. Hutchinson. Visual servo control, Part I: Basic approaches. *IEEE Robotics and Automation Magazine*, 13(4):82–90, December 2006.

[4] F. Chaumette and S. Hutchinson. Visual servo control, Part II: Advanced approaches. *IEEE Robotics and Automation Magazine*, 14(1):109–118, March 2007.

[5] F. Chaumette and E. Malis. 2 1/2 D visual servoing: a possible solution to improve image-based and position-based visual servoings. In *IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 630–635, San Francisco, CA, April 2000.

[6] A.I. Comport, E. Marchand, M. Pressigout, and F. Chaumette. Real-time markerless tracking for augmented reality: the virtual visual servoing framework. *IEEE Trans. on Visualization and Computer Graphics*, 12(4):615–628, July 2006.

[7] L. Coutard and F. Chaumette. Visual detection and 3d model-based tracking for landing on aircraft carrier. In *IEEE Int. Conf. on Robotics and Automation, ICRA'11*, Shanghai, China, May 2011.

[8] F. Dionnet and E. Marchand. Robust stereo tracking for space robotic applications. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'07*, pages 3373–3378, San Diego, CA, October 2007.

[9] T. Drummond and R. Cipolla. Real-time visual tracking of complex structures. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(7):932–946, July 2002.

[10] W. Fehse, editor. *Automated rendezvous and docking of spacecraft*. Cambridge Aerospace Series. Cambridge Univ. Press, 2003.

[11] P. Jasiobedzki, M. Greenspan, and G. Roth. Pose determination and tracking for autonomous satellite capture. In *Proc. of the Int. Symp. on Artificial Intelligence and Robotics & Automation in Space, i-SAIRAS'01,*, volume 15, pages 6–9, Montreal, Canada, 2001.

[12] D.G. Lowe. Fitting parameterized three-dimensional models to images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(5):441–450, May 1991.

[13] D. Pinard, S. Reynaud, P. Delpy, and S.E. Strandmoe. Accurate and autonomous navigation for the ATV. *Aerospace Science and Technology*, 11(6):490–498, September 2007.

[14] D. Tingdahl, M. Vergauwen, and L. Van Gool. Harvd image processing algorithms detailed design. EADS Astrium, 2010.

[15] D.C. Woffinden and D.K. Geller. Navigating the road to autonomous orbital rendezvous. *Journal of Spacecraft and Rockets*, 44(4):898–909, 2007.

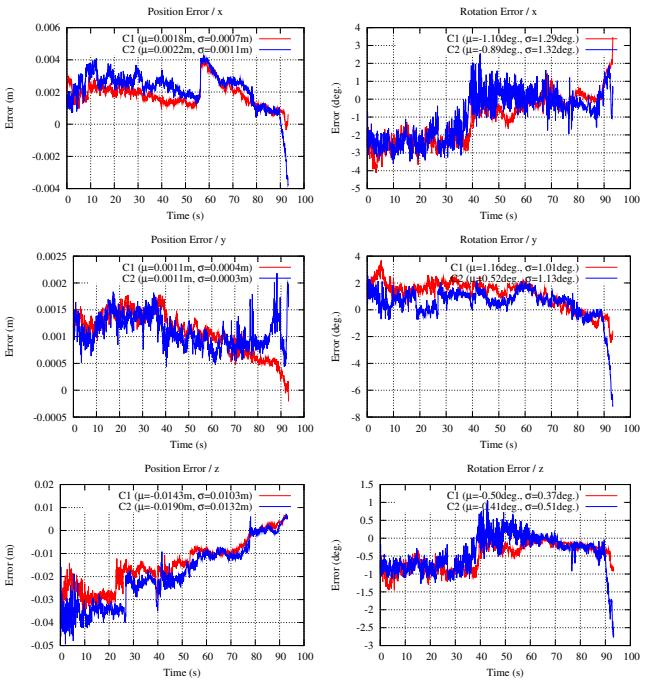


Fig. 10: Pose errors, for the nominal approach scenario, using tracking configurations C1 and C2, with the corresponding mean (μ) and standard deviation (σ) values.

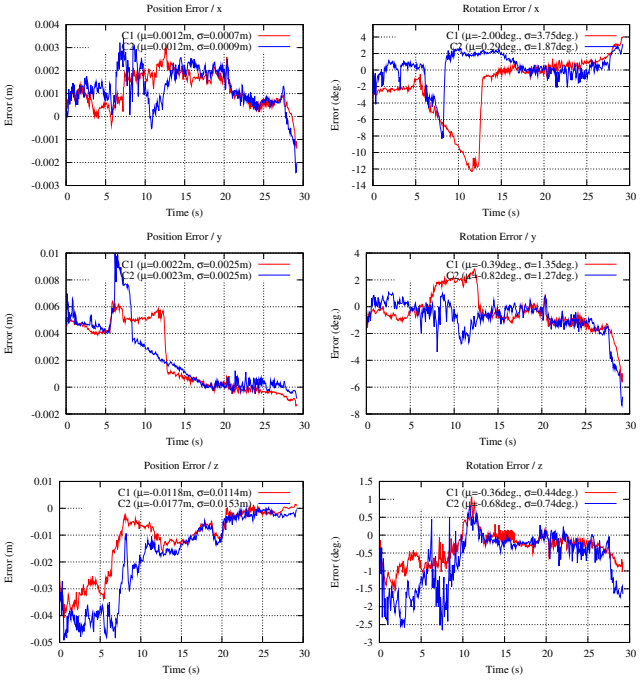


Fig. 11: Pose errors, when the chaser starts its first translation with solar panels of the target appearing as single edges, using configurations C1 and C2.

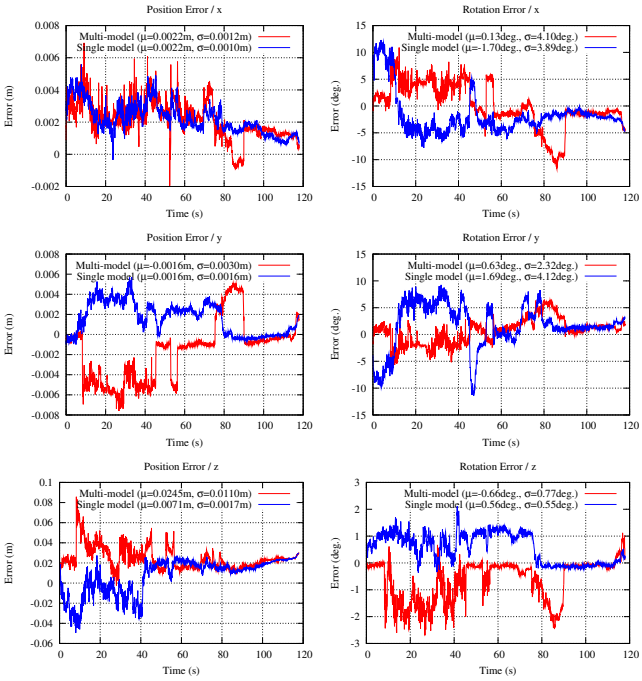


Fig. 12: Pose errors, for a standard approach with six brutal illumination changes, at times $t = 7s$, $45s$, $52s$, $57s$, $74s$ and $88s$, using a configuration with two 3D models switching at illumination changes (Multi-model), and a configuration with a single 3D model during the whole approach.